DTU Library

An End-to-end Deep Learning Approach to MI-EEG Signal Classification for BCIs

Dose, Hauke; Møller, Jakob Skadkær; Iversen, Helle K.; Puthusserypady, Sadasivan

Accepted Manuscript

An End-to-end Deep Learning Approach to MI-EEG Signal Classification for BCIs

Hauke Dose, Jakob S. Møller, Helle K. Iversen, Sadasivan Puthusserypady
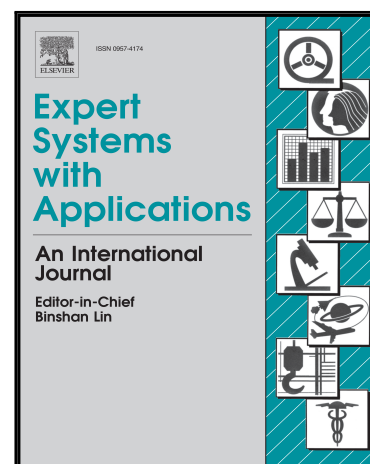
Please cite this article as: Hauke Dose, Jakob S. Møller, Helle K. Iversen, Sadasivan Puthusserypady, An End-to-end Deep Learning Approach to MI-EEG Signal Classification for BCIs, *Expert Systems With Applications* (2018), doi: https://doi.org/10.1016/j.eswa.2018.08.031

Highlights

- End-to-end neural network model for classifying motor imagery EEG signals

- Using 1-D CNN layers to learn temporal and spatial filters for feature extraction

- Application of transfer learning to calibrate the model for individual subjects

- Analysis of the temporal and spatial filters learned by the model

1

An End-to-end Deep Learning Approach to MI-EEG Signal Classification for BCIs

Hauke Dose[a], Jakob S. Møller[a], Helle K. Iversen[b], Sadasivan Puthusserypady[a,*]

[a]*Department of Electrical Engineering, Technical University of Denmark,
Kongens Lyngby, 2800, Denmark*
[b]*Neurological Department, Glostrup Hospital,
Glostrup 2600, Denmark*

Abstract

Goal. To develop and implement a Deep Learning (DL) approach for an electroencephalogram (EEG) based Motor Imagery (MI) Brain-Computer Interface (BCI) system that could potentially be used to improve the current stroke rehabilitation strategies.

Method. The DL model is using Convolutional Neural Network (CNN) layers for learning generalized features and dimension reduction, while a conventional Fully Connected (FC) layer is used for classification. Together they build a unified end-to-end model that can be applied to raw EEG signals. This previously proposed model was applied to a new set of data to validate its robustness against data variations. Furthermore, it was extended by subject-specific adaptation. Lastly, an analysis of the learned filters provides insights into how such a model derives a classification decision.

Results. The selected global classifier reached 80.38%, 69.82%, and 58.58% mean accuracies for datasets with two, three, and four classes, respectively, validated using 5-fold crossvalidation. As a novel approach in this context, transfer learning was used to adapt the global classifier to single individuals improving the overall mean accuracy to 86.49%, 79.25%, and 68.51%, respectively. The global models were trained on 3s segments of EEG data from different subjects than they were tested on, which proved the generalization performance of the model.

Conclusion. The results are comparable with the reported accuracy values in related studies and the presented model outperforms the results in the literature on the same underlying data. Given that the model can learn features from data without having to use specialized feature extraction methods, DL should be considered as an alternative to established EEG classification methods, if enough data is available.

Keywords: Deep Learning (DL), Electroencephalogram (EEG), Motor Imagery (MI), Convolutional Neural Networks (CNNs), Brain Computer Interface (BCI), Stroke Rehabilitation

*Corresponding author
 Email addresses:* hauke.dose@rwth-aachen.de (Hauke Dose), jaskmo@elektro.dtu.dk (Jakob S. Møller),
helle.klingenberg.iversen@regionh.dk (Helle K. Iversen), spu@elektro.dtu.dk (Sadasivan Puthusserypady)

1. Introduction

Stroke is one of the leading causes of adult disability leaving a large part of those surviving the incident with some form of hemiparesis or hemiplegia, which brings a heavy burden to the patient, family and healthcare systems (World Stroke Organization (WSO), 2016). While conventional therapy focuses on physiotherapy and repetitive training for functional recovery (National Institute of Neurological Disorders and Stroke, 2014), new therapy forms are sought for in order to promote regaining independence in daily activities. Different strategies ranging from neuromodulatory techniques (such as repetitive Trans-cranial Magnetic Stimulation (TMS) (Pinter & Brainin, 2013; Bates & Rodger, 2015) or Trans-cranial Direct Current Stimulation (TDCS) (Leuthardt et al., 2009)), to stem cells and pharmacological therapy (Chaudhary et al., 2016), to the usage of Brain-Computer Interface (BCI) systems have emerged to tackle the limitations of current stroke rehabilitation strategies/methods.

BCIs have received a lot of attention recently. Controlling an electric wheelchair (Galán et al., 2008), text input (Guan et al., 2004), or neural bypasses to control paretic limbs (Young et al., 2014) are some of the use cases that demonstrate their possibilities. Motor imagery (MI) (i.e. imagining the execution of movements) has shown to activate similar brain pathways as in actual execution of movements. It shows promise to act as substitute exercises in situations where there is no residual motor function, or to prevent exhaustion (Lotze & Halsband, 2006; Stippich et al., 2002; Mulder, 2007).

Several studies (Meng et al., 2008; Jure et al., 2016; Vallabhaneni & He, 2004; Pichiorri et al., 2015; Tabar & Halici, 2017; Shen et al., 2017; Kim et al., 2016a; Loboda et al., 2014; Park et al., 2013; Do et al., 2011, 2012; Ma et al., 2016; Kumar et al., 2016; Schirrmeister et al., 2017) have investigated the classification of such MI-EEG signals to create a BCI that provides feedback during MI training. Providing proper feedback, possibly through Functional Electric Stimulation (FES) devices is believed to promote neural plasticity and enhance the learning process. This could ultimately lead to faster and better recovery for hemiparetic stroke patients.

The dominating approach for the task of MI-EEG classification is the Common Spatial Patterns (CSP) algorithm (Wang et al., 2005; Grosse-Wentrup & Buss, 2008) and variations thereof, like Common Spatio-Spectral Patterns (CSSP) (Aghaei et al., 2016), Filter Bank CSP (FBCSP) (Ang et al., 2008; Bentlemsan et al., 2014), or Strong Uncorrelating Transform Complex CSP (SUTCCSP) (Park et al., 2014a,b; Kim et al., 2016b). These algorithms extract features from the EEG signals, which are then classified by supervised machine learning algorithms like Support Vector Machines (SVMs). To improve the results, specialized pre-processing, such as linear filtering, artifact removal (Halder et al., 2007), and trial rejection is often applied.

While CSP methods focus heavily on frequency selection, it was recently shown that accounting for temporal variability using time-frequency selection can be beneficial to accuracy and can help reduce the needed amount of EEG channels (Yang et al., 2014, 2016, 2017). This calls for new methods, that detect time-dependent features in EEG signals.

Deep learning (DL) methods are conquering many domains by outperforming conventional methods in a multitude of problems (LeCun et al., 2015). Especially in the domain of image recognition, Convolutional Neural Networks (CNNs) have lead to a significant performance increase almost halving the error rates of competing al-

3

gorithms in the ImageNet competition 2012 (LeCun et al., 2015). Only recently, DL was also applied to EEG classification. Kumar et al. (2016) and Yang et al. (2015) suggested replacing commonly used classifiers like SVMs by Multilayer Perceptrons (MLP) while keeping the specialized feature extraction mechanisms. Bashivan et al. (2016) used CNNs to classify EEG signals through spectral topography maps generated from short-time Fourier transformed (STFT) recordings. Finally, Tabar & Halici (2017) used time-frequency maps from STFT as input to a CNN with stacked auto-encoders (SAE) reaching very high accuracy compared to benchmark methods.

While the approaches applied in Kumar et al. (2016); Bashivan et al. (2016); Tabar & Halici (2017) involve pre-processing, such as the feature extraction or STFT for time-frequency mapping, Shen et al. (2017) proposed an end-to-end DL approach using CNNs and Long Short-Term Memories (LSTMs) to classify raw EEG data. Tang et al. (2017) developed a CNN architecture with one-dimensional convolutional layers to classify raw EEG data. A similar model was developed by Schirrmeister et al. (2017) with CNN input stages for separated temporal and spatial filters yielding exceptional results despite a very simple and shallow architecture.

In the following, we will present a similar architecture and apply it to the Physionet database (AL et al., 2013) of MI recordings. This database contains more samples than the BCI Competition IV 2a dataset used in Schirrmeister et al. (2017) which may improve the results. As a novel approach, we extended the training process to apply transfer learning, which allows for adaptation of the model to a single subject. This improves the final result by accounting for subject-specific differences at the cost of an additional re-training for each subject. To gain insight into the models' learning process, we conclude with a brief analysis of the CNN filters after training.

2. Background

This section introduces the methods that were used to build an MI-EEG classification model. It presents the concept of feed-forward neural networks and the specialized case of the CNN as well as implementation details for training the models effectively.

2.1. Deep feed-forward neural networks

Neural networks are biologically inspired mathematical models, which are conditioned to solve problems that can be generalized to producing an output of a fixed dimension from an input of a fixed dimension. They are a subset of a generalized graph model, where the nodes of the graph are organized in layers. Each node, also referred to as neuron, performs a (possibly non-linear) computation on its inputs and passes the result to the next layer. There, the neurons receive all inputs from the outputs of the previous layer and perform themselves a computation on these.

An artificial neuron describes a function $f : \mathbb{R}^N \to \mathbb{R}$ that computes an output (activation) as the result of a scalar activation function $\sigma : \mathbb{R} \to \mathbb{R}$ evaluated at the weighed sum $b_i + \sum_j w_{ij} x_j$ of the inputs $x \in \mathbb{R}^N$ and a bias b_i. Hence, the activation a_i of a neuron given the activation of the input neurons a'_j is computed as

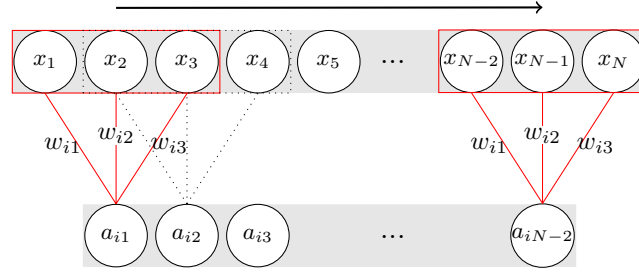$$a_i = \sigma \left(b_i + \sum_j w_{ij} a'_j \right), \tag{1}$$

4

Figure 1: Temporal convolution. A convolutional window of size $k = 3$ moving across a 1D time-series. The three corresponding weights $(w_{i1},\ w_{i2},\ w_{i3})$ for the ith filter are shown and the bias is left out for clarity. Here, N represents length of the input vector $\mathbf{x}_j$.

where w_{ij} is the weight of the connection from the jth neuron of the previous layer to the ith neuron of the layer at hand. Commonly used σ are the rectified linear units (ReLU), hyperbolic tangent or the sigmoid function.

2.2. Convolutional neural networks

Increasing the number of layers *proportionally* increases the number of parameters in a feed-forward model. Even worse, increasing the number of neurons per layer increases the number of parameters *quadratically*, as each neuron in one layer has to be connected to every neuron in the following layer. The resulting high parameter counts therefore disqualify feed-forward models for high-dimensional data like EEG signals of sufficient recording lengths.

This issue is tackled by CNNs which formulate the convolution operation in the neural network context. The convolution of two discrete signals x_n and w_n is defined as $x_n \otimes w_n = \sum_{m=-\infty}^{\infty} x_m w_{n-m}$, where $\otimes$ denotes the convolution operation, this operation can easily be extended to 2-D. Each neuron in the first hidden layer of the CNN is connected only to a small region of the input neurons, known as the convolutional window (Refer Fig. 1). Each connection learns a weight, and the neuron also learns an overall bias. Next, the window is slid across the entire input sequence, and each neuron in the hidden layer learns to analyze a specific part of the input sequence as shown in Fig. 1. The size or length of the convolutional window is known as the kernel size, k. Now instead of learning new weights and biases for each neuron in the hidden layer, the CNN learns only one set of weights and a single bias, which is applied to all neurons in the hidden layer (concept of weight sharing). Mathematically, this can be expressed as:

$$a_{ij} = \sigma \left(b_i + \sum_{k=1}^{3} w_{ik} x_{j+k-1} \right) = \sigma \left(b_i + \mathbf{w}_i^T \mathbf{x}_j \right), \tag{2}$$

where T denotes the transpose operation, a_{ij} is the activation or output of the jth neuron of the ith filter in the hidden layer, b_i is the shared overall bias of filter i, $\mathbf{w}_i = [w_{i1}\ w_{i2}\ w_{i3}]^T$ is a vector with the shared weights and $\mathbf{x}_j = [x_j\ x_{j+1}\ x_{j+2}]^T$. All neurons in the first hidden layer are hence trained to detect the same feature, just at different locations in the input sequence. Because of this property, the activations of the hidden layer is commonly referred to as a feature map (Nielsen, 2015a). Given a finite kernel size (dimension of $\mathbf{w}_i$), the input to a certain neuron only depends on the values of a smaller amount of neurons from the previous layer (sparse connections).

Through these concepts, sparse connections and weight sharing, constraints are imposed on the optimization, which dramatically reduces the amount of weights that have to be learned. This speeds up the training process as

less gradients have to be computed, which is decisive for the computational cost of DNN training. Furthermore, the drastically reduced number of parameters is a natural protection against overfitting, while still granting ability to learn complex features.

The hidden layers in a CNN carry multiple filtered versions of the input data, where certain aspects are pronounced in each of the filtered representations. These representations are often called feature maps. Through stacking of CNN layers, more and more complex features can be learned by the network.

Through pooling operations, the dimensions of an initially high-dimensional input are often incrementally reduced between the layers. This is done by aggregating neighboring values in a feature map into one value by determining the average, maximum, or other operations. This can also increase location independence of the model, as the same features in slightly different locations can ultimately be mapped to the same internal feature due to the aggregation of neighboring neurons.

2.3. Neural network training

Finding suitable weight matrices and bias vectors for the neural network model is typically done through an iterative gradient descent optimization. The efficient computation of the gradient of the cost function (quantifying the deviation of the observed output from the desired output) with respect to every weight is based on the backpropagation algorithm, the derivation of which is omitted here for brevity. We refer to Nielsen (2015b) and Goodfellow et al. (2016) for details.

For this work, Adam was used as the optimization algorithm, which builds upon Stochastic Gradient Descent (SGD) with an adaptive learning rate based on the average first and second order moments of the gradients. This method generally promotes a faster convergence of the model and is more robust in case of noisy and sparse gradients.

3. Method

With the required DL models defined, we will now describe how they were applied to the specific problem of EEG signal classification. CNNs are often used for image classification, which is a task of classifying two-dimensional data. Multichannel EEG data is also two-dimensional, but the two dimensions time and channel have different units, which motivates a non-trivial choice of the filter kernel dimensions.

In this study, the direction of filtering was fixed for each layer resulting in one layer performing a 1-D convolution along the time axis, and a second layer performing a spatial convolution only along the channel axis. The model was applied to the Physionet MI dataset, which was chosen because of its size containing data from 109 subjects. This proved to be large enough to train the proposed model without overfitting too early.

3.1. Model architecture

The model is based on a shallow CNN proposed in Schirrmeister et al. (2017) and consists of two convolutional layers with 40 kernels per layer. The first layer only performs convolution along the time axis. This corresponds

6

Table 1: CNN model architecture: N refers to the length of the input and N_{EEG} to the number of EEG channels used. The parameter counts are provided for two class classification of 6s of EEG data from $N_{\text{EEG}} = 64$ channels.

	kernel	padding	params	output shape
Conv 40	30×1	same	1240	$N \times N_{\text{EEG}} \times 40$
Conv 40	$1 \times N_{\text{EEG}}$	valid	102440	$N \times 1 \times 40$
Avg. pool	15×1	valid	0	$\frac{N}{15} \times 1 \times 40$
Flatten	-	-	0	$\frac{40N}{15}$
FC 80	-	-	201680	80
Softmax	-	-	162	2
			305522	

to a linear pre-filtering of every channel's EEG signal. The second layer only performs convolution along the EEG channel axis, that is, only the different channel values at the same time instances are considered.

The valid padding in the second layer means that the input data is not padded at the outer bounds such that the
120　kernel does not move past the data bounds. With the kernels being equally as long as the number of channels, this operation corresponds to a linear combination of all channel values at a particular instant as there is only one valid position of the kernel. It reduces the channel dimension to one value, such that only 1-D time series are produced as the output of each kernel of the CNN. Subsequently, the output is reduced in an average pooling layer, flattened to 1-D and finally fed into a fully connected (FC) layer. The output is produced by a softmax layer with a number
125　of neurons corresponding to the number of classes in the data to classify for. The softmax operation ensures that the output values sum to one and can therefore be seen as probability/certainty values of the input belonging to a certain class.

Table 1 summarizes the proposed neural network architecture. The number of kernels and kernel dimensions were empirically determined based on the values used in Schirrmeister et al. (2017). Except for the output layer,
130　ReLUs were chosen as activation functions. They have the advantage of a steep gradient even at high deviations from the optimum, which speeds up the optimization process.

With the one-hot output encoding (the target output is a vector with one value per class, which is one for the target class and zero otherwise), the loss function was chosen to be the categorical cross-entropy, which is defined as

$$C(p, q) = -\sum_i p_i \log q_i, \tag{3}$$

where p is the target distribution and q is the observed distribution. In the multi-class scenario, this reduces to $C(p, q) = -\log q_{true}$, where q_{true} is the observed value of the target classes neuron, as $p_i = 0$ for all other terms of the summation.

135　Table 2 provides an overview over the remaining design decisions and hyperparameters. Their choice is based on empirical trails and choices made in related works (Schirrmeister et al., 2017; Shen et al., 2017).

To sum up, Fig.2 illustrates the network's conceptual architecture. The different layers of the network can be interpreted as a typical signal processing pipeline. First, a filter bank of 40 FIR-filters pre-filters the time series.

Table 2: Model hyperparameters

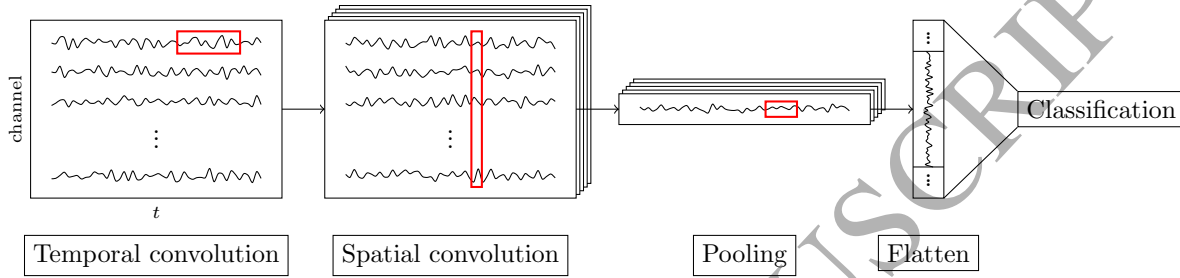| parameter	value
optimizer	Adam
activation	ReLU
regularization	0
cost function	categorical cross-entropy
batch size	16

Figure 2: General architecture: The layers of the CNN can be described as different types of filtering and processing steps before the actual classification by the FC/softmax layers. The red rectangles indicate the filter kernel and pooling dimensions.

The filter coefficients are learned automatically, but likely result in filters for frequency separation. Then, the

140 filtered signals are combined by spatial filters. In this layer, different linear combinations of frequency contents and channel locations are learned, which achieve a high inter-class difference to facilitate the classification through the FC/Softmax layers.

Ideally, the model will learn to ignore artifacts that are not relevant to the classification result in the course of the optimization process. It was deliberately chosen not to apply any artifact removal or filtering beforehand

145 (Halder et al., 2007), to emphasize the end-to-end capability of the model.

3.2. Overfitting

As stated in Table 2, the regularization term is set to be zero as neither dropout nor L1 or L2 regularization showed to improve the generalization performance. On the other hand, even without regularization, the model was not prone to overfitting unless the amount of input data was reduced significantly. This shows that the model might

150 well be applicable to smaller datasets.

The reason for the robustness against overfitting may be due to the drastic dimension reduction through the model's pooling layers and the batch learning process, which averages over 16 samples from different subjects in each training step.

8

4. Data

¹⁵⁵ This section introduces the data used for training and evaluation of the proposed method. We also describe the applied method of splitting data into training and test data.

4.1. The Physionet database

The methods presented in the previous section have been applied to the Physionet EEG Motor Movement/MI Dataset (AL et al., 2013), which was recorded by the developers of the BCI2000 system (Schalk et al., 2004). The ¹⁶⁰ dataset[1] contains both recordings of motor execution tasks (i.e. actual execution of movements), as well as MI tasks from the same subjects. Only the MI trials were used in this work.

There are recordings from 109 different subjects, where each performed three two-minute runs of two different MI tasks (left/right fist *or* both fists/both feet). The data is publicly available as raw time series data from 64 EEG channels sampled at 160Hz. The amount of data makes it particularly well suited for DL models, as these are ¹⁶⁵ known to excel at the presence of large amounts of training data.

4.2. Paradigm

The experimental paradigm is illustrated in Fig.3. The subject sits relaxed in front of a computer screen. The trial starts at $t = -2s$. At $t = 0s$ a target appears and its position on the screen dictates the movement to be imagined. It appears left or right for the task of opening and closing the left or right fist, respectively. In another ¹⁷⁰ task, it appears at the top or the bottom indicating the movement of either both fists or both feet. The target remains on the screen for 4s during which the subject is asked to execute the movement. When the target disappears at $t = 4s$, the subject relaxes. After additional 2s, a new trial begins.

4.3. Data subsets

The following subsets were created from the available data.

¹⁷⁵ - **2-class**: MI trials of opening and closing either the right or the left fist. Due to some variability in the number of trials per subject, a subset of 105 subjects and 42 trials per subject were selected (21 for each side), although most subjects performed more than 42 trials.

- **3-class**: Random sections from the available baseline recordings with eyes open were added to the 2-class data to obtain a total of 63 trials per subject with 21 trials per class. This third class represents the resting ¹⁸⁰ state, where the subject is not performing any MI task.

- **4-class**: The fourth class corresponds to MI of both feet, which was performed in the sessions where the subjects were asked to perform imagined movements of either both fists or both feet. The trials, where both fists moved were discarded as they were expected to share features with the single fist trials. This set contains 84 trials per subject with 21 trials per class.

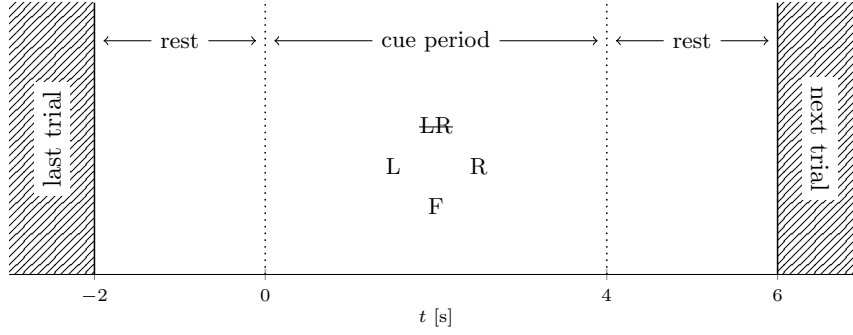[1]publicly available at `https://doi.org/10.13026/C28G6P`

Figure 3: Experimental paradigm used in the Physionet EEG dataset. After a resting period the subject is cued to execute an MI task for four seconds. The class both fists (LR) was not used as the task overlaps with the single fist classes L and R.

4.4. Crossvalidation

To obtain valid results, the data was separated into five 80/20 crossvalidation splits by subject. That is, every model was trained five times on 80% of the subjects and tested on the remaining 20% of the subjects. This ensures the applicability of results to new cases as the global model was not trained on any data from the subjects it was tested on. If not mentioned otherwise, the mean maximum accuracy across all five splits is reported as a global measure.

5. Results

In this section, the results achieved by applying the proposed model to the different data subsets are presented.

5.1. Offline global model performance

The first approach that was considered was a model classifying full trials of 6s (one second rest before and after four seconds of MI) with data from all 64 EEG channels. This approach can make use of all available information and is expected to deliver the highest classification performance. The model reached an average crossvalidation accuracy of **87.98%**, **76.61%**, and **65.73%**, respectively for the two-, three-, and four-class classification tasks. Reducing the amount of input data to the first three seconds after the MI cue delivered global accuracy values of **80.38%**, **69.82%**, and **58.58%**, respectively. The significantly lower performances are likely a result of additional information from the end of the MI period, which this model could not make use of.

To examine the results of the first model in greater detail, Fig. 4 presents scatter plots of the first two principal components of a test set at the output of the FC layer, that is the layer just before the softmax classifier. It can be used to interpret the quality of the learned feature detector. A good feature detector should produce a large inter-class variance. Generally, we can state that there are no significant outliers in this example. This shows that the model is able to obtain generalized features, which correspond to the class differences, but they are partly too similar to be classified correctly.

10

2-class

variance explained: 86.38%

3-class

variance explained: 76.43%
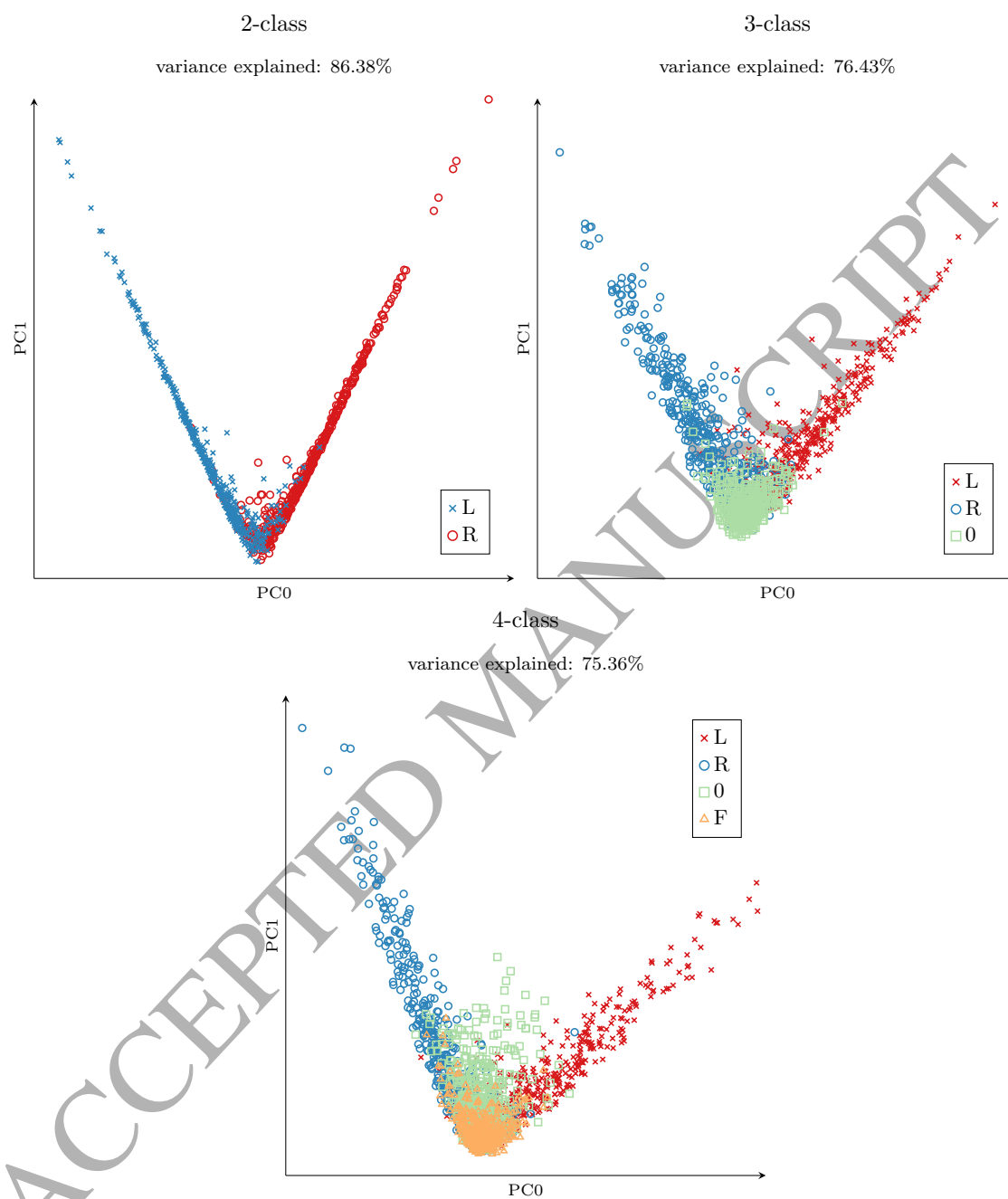
4-class

variance explained: 75.36%

Figure 4: The first principal components at the output of the FC layer. The classes L and R are well separated, but they overlap where class 0 and F are located, which makes them harder to classify.

11

Table 3: Confusion matrix for 4 class classification. Many L, R, and F samples are classified as 0 leading to class 0 featuring the highest recall, but the lowest precision.

		predicted				
		L	R	0	F	recall
actual	L	325	16	100	42	0.672
	R	15	287	120	61	0.594
	0	28	46	358	51	0.741
	F	39	16	96	332	0.687
precision		0.799	0.786	0.531	0.683	

The two-class network shows a good separation of the test data, where two long clusters of the actual classes are immediately visible. They partly overlap, which is where we expect cases that are harder to classify, possibly because the subject did not produce significant MI brain signals.

The resting state 0 and class F are found around the intersection as well, which increases difficulty even more. The expected consequences can be confirmed by the confusion matrix associated with the same test set presented in Table 3. As can be expected from Fig. 4, many samples are falsely classified as 0 leading to a low precision for this class. In turn, its recall is high, as there are fewer trials with true class 0, which are classified as something else.

This leads to the conclusion that the model tends to classify samples as 0, if less distinct EEG activity can be extracted from the signal. These are most likely cases that are located in the intersection area in the PCA plots. Intuitively, this makes sense, as trials with less distinctive MI features may look like the subject is resting and not performing any task at all. With respect to a system with real-time stimulation of muscles, this is desirable. If the model is uncertain, it should rather tend to output class 0 instead of a wrong guess, which would trigger an undesired feedback.

5.2. Adapting the global model

It is a reasonable assumption that different subjects can yield a different individual accuracy, because they might be more or less able to perform MI and activate the detectable brain circuits. Another reason, however, could be individual features in the EEG signal, which are not representative for all subjects, but actually facilitate classification. In this case, adapting the global classifier to a single person can improve the performance of a classifier for that individual.

As a novel approach compared to the previous work of Schirrmeister et al. (2017), transfer learning is applied to yield a subject-specific classifier. First a global model was trained as before. Second, for each subject in the test set, the trials were split into four 75/25 crossvalidation splits and the model was trained for five more epochs. This approach only requires one global model to be trained. The subsequent adaptation of each subject's model only requires a comparatively short training period and less data. Figure 5 shows the distributions of subject-based classification accuracy on the four class set before and after the subject-specific training step.

Apart from an apparent shift of the distribution towards higher accuracy, the distribution also got narrower
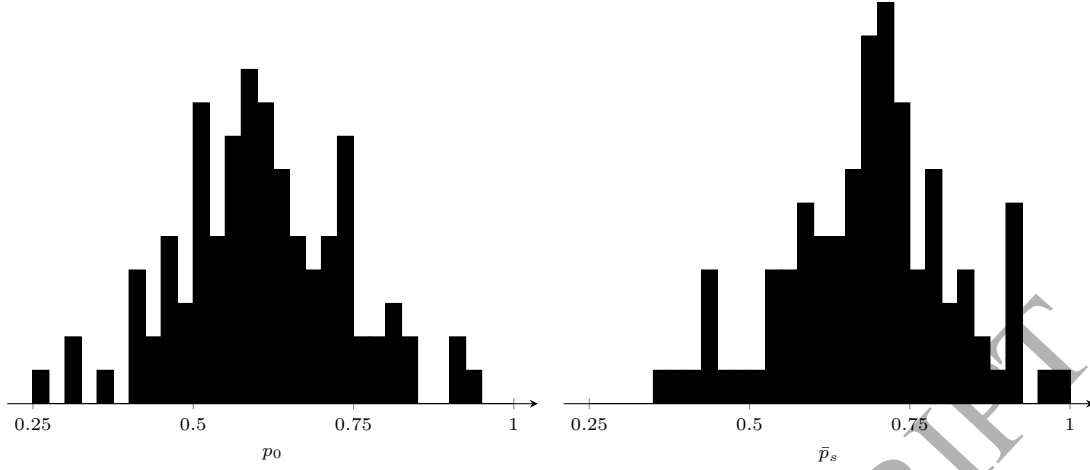
Figure 5: The empirical distribution of subject-specific test accuracy for the four-class task. *Left*: global classifier (subject-independent). *Right*: subject-specific classifier adapted on subject data.

Table 4: Mean accuracy values $\bar{p}_0$ (global classifier) and $\bar{p}_s$ (subject-specific classifiers) and their standard deviation between subjects for a classifier using 64 channels and 3s input segments.

	$\bar{p}_0$	σ_0	$\bar{p}_s$	σ_s	$\bar{p}_s$ - p_0
2 classes	0.8038	0.1254	0.8649	0.1009	+6.11%
3 classes	0.6982	0.1432	0.7925	0.1066	+9.43%
4 classes	0.5859	0.1467	0.6851	0.1256	+9.92%

(standard deviation 0.1256 vs. 0.1467 before retraining). This suggests, that the adaptation of the model has accounted for some of the inter-subject variability in the performance. Finally, Fig.6 shows the distributions of the improvement ($\bar{p}_s - p_0$) for the three sets of EEG data. It is evident that only a few subjects did not yield an improved accuracy in each of the sets. Several subjects even attain an up to 30% increased accuracy value.

Table 4 states the average improvement over all subjects. While the two class accuracy improved by about 6%, over 9% improvement was reached for the three and four class sets. On average, "calibrating" the model on subject-specific data can therefore be a viable measure to increase the performance, if the additional effort of calibrating the model for the subject is an option.

5.3. Online global model performance

Previously, the models presented were trained and tested on a fixed segment of every trial (e.g. the first three seconds after the cue). This is a valid measure when the offline performance shall be determined. In turn, if a real-time system is considered, that is supposed to provide feedback to the patient while performing the exercise, the presented approach will likely only achieve good results in the short part of the trial it was trained on. To resolve this, a model was trained on segments of 2.5s with a random offset. The selection was constrained such that at least half of the 2.5s were located after the cue and before the end of the cue. This ensures that there is actually MI activity in the training samples.
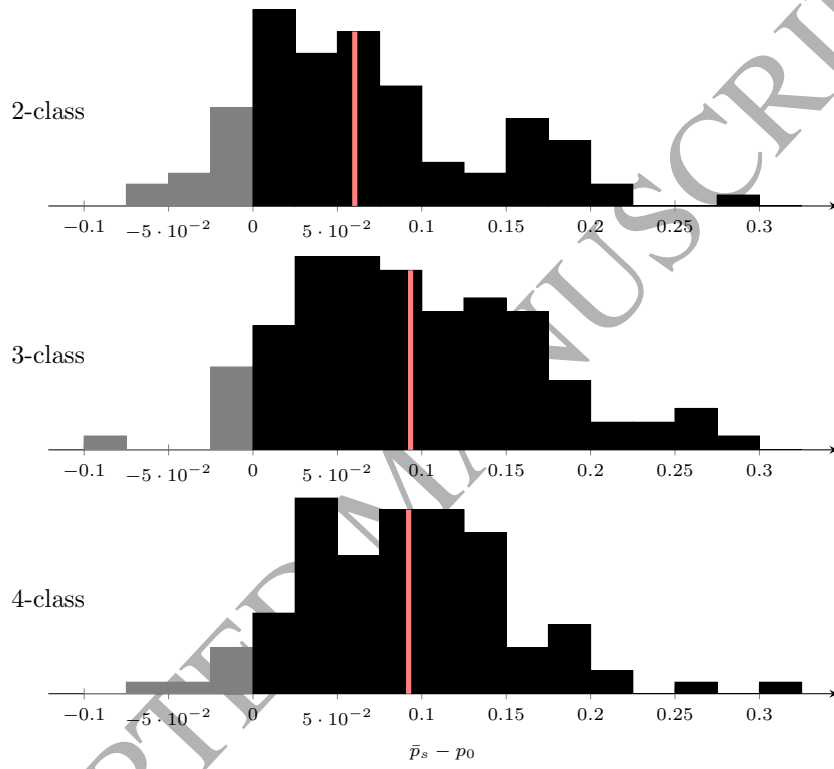
13

Figure 6: Distributions of accuracy improvement $\bar{p}_s - p_0$ after retraining with subject data. Only few subjects do not benefit from retraining. The mean improvement (vertical line) is higher for the three and four class data compared to the two class data.
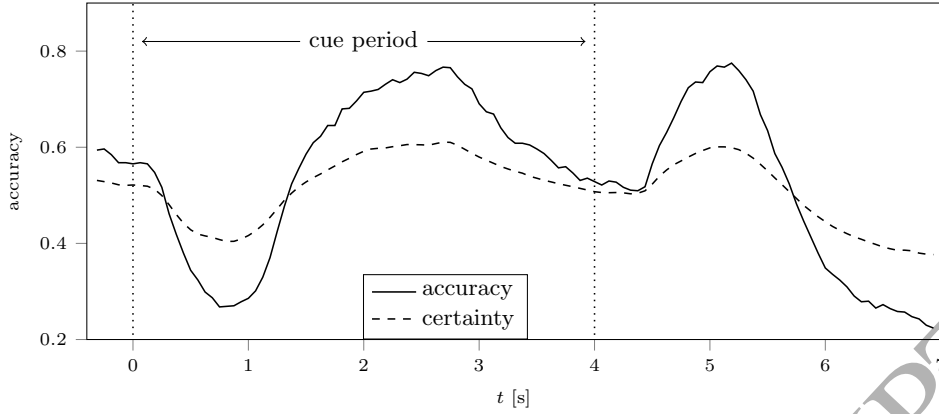
Figure 7: Accuracy and mean certainty y_{true} of a two class classifier over time offset (relative to the cue) of the test input sample.

The trained model was then validated with fixed time offsets to verify the performance over time. The result is shown in Fig. 7 along with the actual certainty p_{true} of the classifier. The time-axis is adjusted such that the accuracy value at time t represents the accuracy on the test set with EEG data from the interval $[t - 2.5s, t]$.

At the beginning, the accuracy lies around the chance level of 0.5. Then, after the cue is first displayed, the accuracy drops significantly below chance level. This means that when a subject starts an MI task, the model interprets the movement onset as MI activity of the opposite class and more than 70% of the trials are falsely interpreted. When the MI activity continues, the accuracy rises to its maximum at about 2.5 seconds after cue onset. This corresponds to the input length of the classifier, so at this point almost the entire input contains MI activity. Towards the end of the imagined movement, the accuracy decays, before it rises again when the subject is cued to stop the movement. These results suggest, that the classifier is better in classifying the start or stop of MI activity than a period of sustained activity.

In general, the certainty correlates with the accuracy, which qualifies it as an actual certainty measure to estimate how much the results can be trusted. The observation that the accuracy drops far below chance level for a certain offset, hints at a general imperfection of the model architecture caused by the fact that the onset of a movement at the end of the input segment is interpreted as the opposite class.

5.4. Analysis of the learned filters

Finally, it is worth analyzing the filters that are learned by the network. This helps understand, which frequency bands and EEG channels the network mainly bases its decision on. In the following, a few illustrative examples have been chosen to give an impression of the filters' properties. The actual results, however, vary significantly between models.

First, we will examine the learned temporal filters. Out of the 40 filters at the input of a three class classifier, five examples were chosen to illustrate different general traits, that were observed. While many filters are low-pass, there are also filters emphasizing a certain frequency range or several frequency ranges at once. The focus was on selecting filters that differ from each other as much as possible to show the found variations. To analyze the
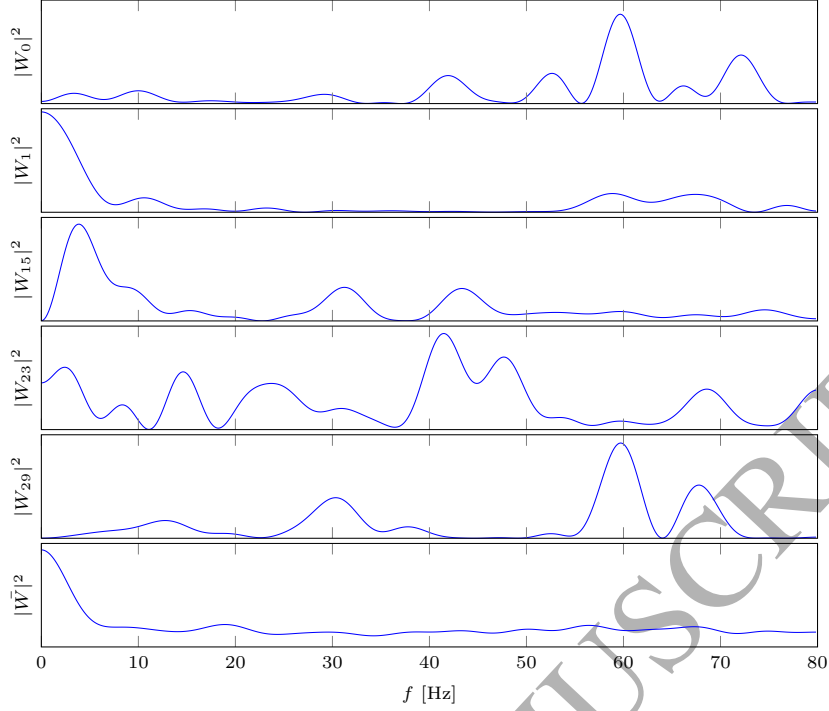
15

Figure 8: Five examples of the squared frequency responses $|W_i(f)|^2$ of the learned temporal filters and the mean $|\bar{W}|^2$ (bottom). Different frequencies are pronounced/attenuated for the different filters. The mean shows a focus on the lower frequencies (below 10 Hz).

frequency dependent behavior of these filters, we will analyze the squared frequency response $|W_i(f)|^2$ given by

$$|W_i(f)|^2 = \left| \sum_{k=1}^{N_t} w_{ik} e^{-j2\pi f(k-1)} \right|^2, \tag{4}$$

275   where N_t is the length of the temporal filter and w_{ik}, the kth filter coefficient of the ith filter as in Eq.(2).

Figure 8 shows the squared frequency responses of the derived filters and the mean squared frequency response over all 40 filters. Most of the filters attenuate a large part of the spectrum, which means that only signal contents in small frequency ranges can pass the filter. While the filters 1 and 15 are focused on low frequencies below 20 Hz, the filters 0, 23, and 29 have pass bands in the area of 40-70 Hz. This observation confirms that the temporal
280   filtering stage can be seen as a frequency separation layer.

The fact that filters with high pass-band frequencies above 40 Hz are also learned by the model shows, that activity related to movement and MI is also detectable in ranges of high frequency. On the one hand, this suggests that low-pass filtering the signal before processing, as it is often done (refer to (Ang et al., 2008; Bentlemsan et al., 2014; Park et al., 2014a; Tabar & Halici, 2017)), may even discard relevant information and degrade the classification
285   results. On the other hand, the mean frequency response $|\bar{W}|^2$ over all filters shows that on average, the higher frequencies are more often attenuated, while frequencies in the area of 0-10 Hz seem to be the most relevant to the classification.
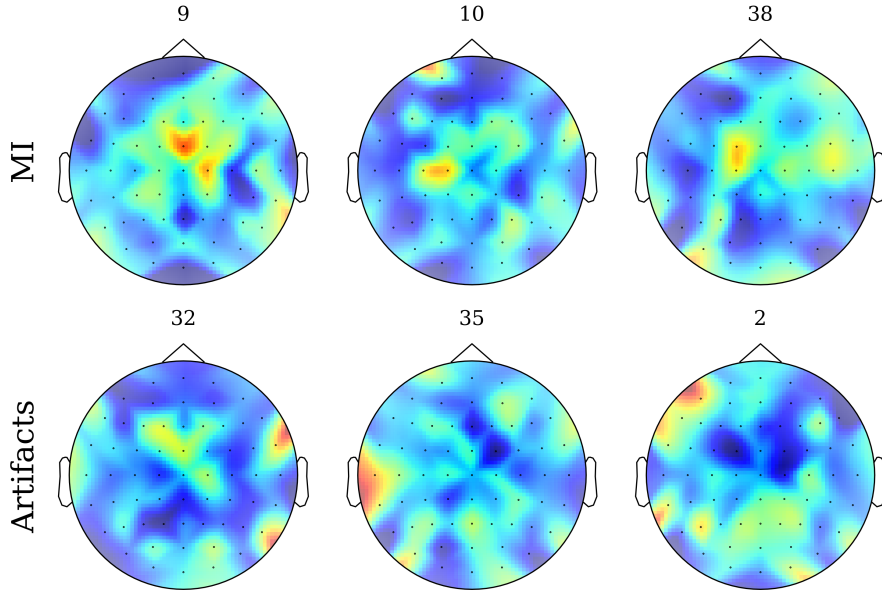
16

Figure 9: Examples of learned spatial filters for three class classification. The upper row shows filters that are assumed to detect MI activity, while the bottom row shows filters, that possibly are used to subtract artifacts like EMG and EOG activity.

The second convolutional layer receives as input 40 differently filtered versions of each input signal. To interpret the filters of the second layer's weights, they were averaged along this dimension to factor out the frequency dimension. Figure 9 shows six examples of the resulting spatial filters. The filters generally appear sparse, i.e. only few adjacent electrodes receive a high weight by the filter. With the task of L/R/0 classification in mind, it is not immediately clear, which filter could act as a feature extractor indicating activity in one hemisphere, as mostly electrodes on both sides of any particular filter have high average weights.

The spatial filters are not easy to interpret and they do not resemble the results typically achieved in CSP-based approaches where the filters tend to pronounce areas of adjacent electrodes (Ang et al., 2008; Park et al., 2014a; Wang et al., 2005). One particular reason for this may be that the shown filters are actually each an average of 40 filters for differently filtered versions of a signal. This may cause less distinct filter weights and averaged-out properties not visible in the topography maps.

Another reason might be the fact that the model is actually combining temporal, spatial and frequency information (Yang et al., 2016). This means that the model might through the temporal convolution also be able to reject and emphasize certain time segments, which adds another dimension and makes the filters harder to interpret.

A few possible MI filters are shown in Fig. 9 top row (filters 9,10 and 38). As the model has many filters to train, it possibly learns more fine-grained features, which only result in a reliable result, when they are combined in the following layers of the model. It is also worth noticing that the model seems to learns spatial filters that look like artifact filters for Electromyography (filters 32 and 35 in Fig. 9) and Electrooculography (filter 2 in Fig. 9). This makes sense due to the raw EEG input.

17

5.5. Training and analysis time

The experiments were conducted on a consumer laptop with an Intel Core i7-6700HQ 2.6GHz quad-core CPU, 16 GB RAM, and a Nvidia GeForce GTX 960M GPU with CUDA toolkit 9.0. A full training cycle of five crossvalidation splits over three epochs took: **3m 51s** for 2 classes, **6m 2s** for 3 classes, and **8m 3s** for 4 classes. This low training effort should not have negative implications for the productive application of the model.

The moderate training time would also allow for applying the model to a much larger datasets or data sampled at higher frequencies without rendering the training unfeasible in terms of time consumption.

Evaluating a single 3s trial (480 samples) takes approximately 3ms in this setup. Given the sample rate of 160Hz ($T = 6.25ms$), it would be possible to retrieve a live prediction for each sample based on the previous 480 samples in an online scenario.

6. Discussion

To compare the performance of the obtained classifier in terms of overall classification accuracy, the results from our two-class model classifying 3s of MI data is used. In Table 5 it is compared with other works that performed the same classification task on the same data. Additional classifiers using the same channel configurations as in the reference works were trained to show the impact of reducing the number of EEG channels on the classification performance and to compare the results.

Our method outperforms the best global and subject-specific models, which shows that it likely represents a true alternative to the conventional methods. That said, as the number of works using the Physionet MI data is limited, we cannot claim that the presented competitors represent the current state of research. Nevertheless, the achieved global accuracy is close to the 80-90% that are commonly achieved on similar data.

Because this method requires more data compared to other methods, it will be necessary to verify if similar results can be obtained on other datasets. As it is difficult to collect large amounts of EEG data in the exact same environment, possibilities to combine datasets and extract generalized features, which are not dependent on the equipment, have to be explored. It is reasonable to assume that, given enough data, a complex enough model will be able to reject equipment-dependent features that are not related to MI activity. Given the flexible architecture of the model, application to other datasets with different channel configurations or sample frequencies would not require major changes to the architecture.

We have also shown in section 5.2, that the presented model can be adapted to single subjects, which in most cases improves the performance. This presents another argument for finding a global model that represents an acceptable fit for a large part of the data, but which can be adapted to single subjects, and possibly also equipment setups or environments, using minimal effort and requiring less data than the complete training process.

In section 5.3, applying the model in a simulated online setting revealed a possible issue with a drop in the accuracy after movement onset. It hints at a general imperfection of the model's architecture, which has to be resolved if the method is to be applied as a real-time classifier.

18

Table 5: Overview over other works performing L/R classification tasks on the Physionet EEG dataset and comparison to this work's results. The same channel configurations (amount and location) as in the reference works were used for comparison.

Work	N_{EEG}	Training	Max. acc.	Methods
(Park et al., 2014b)	58	global	72.37%	SUT-CCSP SVM
(Handiru & Prasad, 2016)	16	global	63.62%	FB-CSP SVM classifier
(Kim et al., 2016b)	14	subject	80.05%	SUT-CCSP Random forest
(Loboda et al., 2014)	9	global	71.55%	Phase information
(Tolic & Jovic, 2013)	3	subject	68.21%	Wavelet transform DNN
This work	64	global	80.38%	CNN
		subject	86.49%	
	58	global	**80.32%**	
		subject	85.55%	
	16	global	**78.03%**	
		subject	83.55%	
	14	global	76.66%	
		subject	**82.66%**	
	9	global	**75.85%**	
		subject	81.82%	
	3	global	73.20%	
		subject	**79.20%**	

19

As there exists no algorithms or fixed guidelines on how to devise an optimal neural network architecture, the presented model can most likely be improved to also improve the results even further. It is worth investigating possibilities of using recurrent neural networks like LSTMs for the sequence classification. These types do not require a fixed input length, which appears more suitable for real-time application with online feedback.

This study has also shown that the predictor draws most relevant information from the beginning and the end of the imagined movement rather than the period in between. A recurrent or other further developed models may be constructed in a way so it can use the presence of a probable movement onset to increase its prediction certainty for the following part of the recording as the movement type never changes abruptly in the given experimental paradigm.

7. Conclusion

The results from these extensive studies indicate that DL can also be applied on EEG signals to create BCI systems for rehabilitative therapy. The applied model was derived from related work, where a similar architecture yielded convincing results. As a novel approach to improve the per-subject performance, it was shown that this model is also adaptable to single subjects through transfer learning. Furthermore, we provided an analysis of the trained parameters in an attempt to understand the decision-making process of the model.

Considering the adaptability of the model and the end-to-end nature that does not require expert knowledge or specialized tools, the approach offers several advantages over specialized methods that are now most commonly used to classify MI-EEG signals.

The classification performance of up to 86.13% in a binary classification task outperformed the best method known to us by around 6%, and remained ahead by more than 2.5% when using the exact same EEG channel configuration for comparison. Future research has to focus on confirming these results across different datasets.

The training process of the model features that it can both be used as a subject-independent classifier, as well as a classifier adapted to a single subject. While showing a good global performance, adapting the model yielded an average improvement of 6-9% in terms of classification accuracy. Adapting the model only requires little data, as it builds upon the previous training results of the global classifier.

We conclude that given enough suitable data and further development towards a good network architecture, deep neural networks could ultimately outperform and substitute the current standard algorithms.

References

References

Aghaei, A. S., Mahanta, M. S., & Plataniotis, K. N. (2016). Separable common spatio-spectral patterns for motor imagery BCI systems. *IEEE Transactions on Biomedical Engineering*, *63*, 15–29. doi:10.1109/TBME.2015.2487738.

AL, G., LAN, A., L, G., JM, H., PCh, I., RG, M., JE, M., GB, M., C-K, P., & HE, S. (2013). Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals. *Circulation*, *101*, e215–e220. URL: http://circ.ahajournals.org/cgi/content/full/101/23/e215.

Ang, K. K., Chin, Z. Y., Zhang, H., & Guan, C. (2008). Filter bank common spatial pattern (FBCSP) in brain-computer interface. In *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)* (pp. 2390–2397). doi:10.1109/IJCNN.2008.4634130.

Bashivan, P., Rish, I., Yeasin, M., & Codella, N. (2016). Learning representations from EEG with deep recurrent-convolutional neural networks. *International Conference on Learning Representations (ICLR)*, .

Bates, K., & Rodger, J. (2015). Repetitive transcranial magnetic stimulation for stroke rehabilitation - potential therapy or misplaced hope? *Restor. Neurol. Neurosci.*, *33*, 557–69.

Bentlemsan, M., Zemouri, E. T., Bouchaffra, D., Yahya-Zoubir, B., & Ferroudji, K. (2014). Random forest and filter bank common spatial patterns for eeg-based motor imagery classification. In *2014 5th International Conference on Intelligent Systems, Modelling and Simulation* (pp. 235–238). doi:10.1109/ISMS.2014.46.

Chaudhary, U., Birbaumer, N., & Ramos-Murguialday, A. (2016). Brain-computer interfaces for communication and rehabilitation. *Nat. Rev. Neurol.*, *12*, 513—525.

Do, A. H., Wang, P. T., King, C. E., Abiri, A., & Nenadic, Z. (2011). Brain-computer interface controlled functional electrical stimulation system for ankle movement. *Journal of NeuroEngineering and Rehabilitation*, *8*, 49. doi:10.1186/1743-0003-8-49.

Do, A. H., Wang, P. T., King, C. E., Schombs, A., Cramer, S. C., & Nenadic, Z. (2012). Brain-computer interface controlled functional electrical stimulation device for foot drop due to stroke. In *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society* (pp. 6414–6417). doi:10.1109/EMBC.2012.6347462.

Galán, F., Nuttin, M., Lew, E., Ferrez, P., Vanacker, G., Philips, J., & del R. Millán, J. (2008). A brain-actuated wheelchair: Asynchronous and non-invasive brain–computer interfaces for continuous control of robots. *Clinical Neurophysiology*, *119*, 2159 – 2169. doi:10.1016/j.clinph.2008.06.001.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

Grosse-Wentrup, M., & Buss, M. (2008). Multiclass common spatial patterns and information theoretic feature extraction. *IEEE Transactions on Biomedical Engineering*, *55*, 1991–2000. doi:10.1109/TBME.2008.921154.

Guan, C., Thulasidas, M., & Wu, J. (2004). High performance P300 speller for brain-computer interface. In *IEEE International Workshop on Biomedical Circuits and Systems, 2004.*. doi:10.1109/BIOCAS.2004.1454155.

21

Halder, S., Bensch, M., Mellinger, J., Bogdan, M., Kübler, A., Birbaumer, N., & Rosenstiel, W. (2007). On-
line artifact removal for brain-computer interfaces using support vector machines and blind source separation.
Computational Intelligence and Neuroscience, *2007*. doi:10.1155/2007/82069.

Handiru, V. S., & Prasad, V. A. (2016). Optimized bi-objective EEG channel selection and cross-subject gen-
eralization with brain-computer interfaces. *IEEE Transactions on Human-Machine Systems*, *46*, 777–786.
doi:10.1109/THMS.2016.2573827.

Jure, F. A., Carrere, L. C., Gentiletti, G. G., & Tabernig, C. B. (2016). BCI-FES system for neuro-rehabilitation
of stroke patients. *Journal of Physics: Conference Series*, *705*. doi:10.1088/1742-6596/705/1/012058.

Kim, Y., Ryu, J., Keun Kim, K., Cheong Took, C., P. Mandic, D., & Park, C. (2016a). Motor imagery classification
using mu and beta rhythms of EEG with strong uncorrelating transform based complex common spatial patterns.
Computational Intelligence and Neuroscience, *2016*.

Kim, Y., Ryu, J., Kim, K. K., Took, C. C., Mandic, D. P., & Park, C. (2016b). Motor imagery classification using
mu and beta rhythms of EEG with strong uncorrelating transform based complex common spatial patterns.
Intell. Neuroscience, *2016*. doi:10.1155/2016/1489692.

Kumar, S., Sharma, A., Mamun, K., & Tsunoda, T. (2016). A deep learning approach for motor imagery EEG
signal classification. In *3rd Asia-Pacific World Congress on Computer Science and Engineering (APWC on CSE)*
(pp. 34–39). doi:10.1109/APWC-on-CSE.2016.017.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*, 436–444. doi:10.1038/nature14539.

Leuthardt, E., Schalk, G., Roland, J., Rouse, A., & Moran, D. (2009). Evolution of brain-computer interfaces:
going beyond classic motor physiology. *Neurosurg. Focus*, *27*.

Loboda, A., Margineanu, A., Rotariu, G., & Mihaela, A. (2014). Discrimination of EEG-based motor imagery
tasks by means of a simple phase information method. *International Journal of Advanced Research in Artificial
Intelligence*, *3*.

Lotze, M., & Halsband, U. (2006). Motor imagery. *Journal of Physiology - Paris*, *99*, 386 – 395.

Ma, Y., Di, X., She, Q., Luo, Z., Potter, T., & Zhang, Y. (2016). Classification of motor imagery EEG signals
with support vector machines and particle swarm optimization. *Computational and Mathematical Methods in
Medicine*, *2016*, 8. doi:10.1155/2016/4941235.

Meng, F., yu Tong, K., tak Chan, S., wa Wong, W., him Lui, K., wing Tang, K., Gao, X., & Gao, S. (2008). BCI-
FES training system design and implementation for rehabilitation of stroke patients. In *2008 IEEE International
Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)* (pp. 4103–4106).
doi:10.1109/IJCNN.2008.4634388.

22

Mulder, T. (2007). Motor imagery and action observation: cognitive tools for rehabilitation. *Journal of Neural Transmission*, *114*, 1265–1278. doi:10.1007/s00702-007-0763-z.

National Institute of Neurological Disorders and Stroke (2014). Post-stroke rehabilitation.

Nielsen, M. A. (2015a). *Neural Networks and Deep Learning*. Determination Press.

Nielsen, M. A. (2015b). *Neural Networks and Deep Learning*. Determination Press. http://neuralnetworksanddeeplearning.com.

Park, C., Looney, D., ur Rehman, N., Ahrabian, A., & Mandic, D. P. (2013). Classification of motor imagery BCI using multivariate empirical mode decomposition. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *21*, 10–22. doi:10.1109/TNSRE.2012.2229296.

Park, C., Took, C. C., & Mandic, D. P. (2014a). Augmented complex common spatial patterns for classification of noncircular eeg from motor imagery tasks. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *22*, 1–10. doi:10.1109/TNSRE.2013.2294903.

Park, C., Took, C. C., & Mandic, D. P. (2014b). Augmented complex common spatial patterns for classification of noncircular EEG from motor imagery tasks. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *22*, 1–10. doi:10.1109/TNSRE.2013.2294903.

Pichiorri, F., Morone, G., Petti, M., Toppi, J., Pisotta, I., Molinari, M., Paolucci, S., Inghilleri, M., Astolfi, L., Cincotti, F., & Mattia, D. (2015). Brain–computer interface boosts motor imagery practice during stroke recovery. *Annals of Neurology*, *77*, 851–865. doi:10.1002/ana.24390.

Pinter, M., & Brainin, M. (2013). Role of repetitive transcranial magnetic stimulation in stroke rehabilitation. *Front. Neurol. Neurosci.*, *32*, 112–21.

Schalk, G., McFarland, D., Hinterberger, T., Birbaumer, N., & Wolpaw, J. (2004). BCI2000: A general-purpose brain-computer interface (BCI) system. *IEEE Transactions on Biomedical Engineering*, *51*, 1034–1043.

Schirrmeister, R. T., Springenberg, J. T., Fiederer, L. D. J., Glasstetter, M., Eggensperger, K., Tangermann, M., Hutter, F., Burgard, W., & Ball, T. (2017). Deep learning with convolutional neural networks for EEG decoding and visualization. *Human Brain Mapping*, *38*, 5391–5420. doi:10.1002/hbm.23730.

Shen, Y., Lu, H., & Jia, J. (2017). Classification of motor imagery EEG signals with deep learning models. In *Intelligence Science and Big Data Engineering: 7th International Conference, IScIDE 2017* (pp. 181–190). Springer International Publishing. doi:10.1007/978-3-319-67777-4_16.

Stippich, C., Ochmann, H., & Sartor, K. (2002). Somatotopic mapping of the human primary sensorimotor cortex during motor imagery and motor execution by functional magnetic resonance imaging. *Neuroscience Letters*, *331*, 50 – 54. doi:10.1016/S0304-3940(02)00826-1.

Tabar, Y. R., & Halici, U. (2017). A novel deep learning approach for classification of EEG motor imagery signals. *Journal of Neural Engineering*, *14*, 16003. doi:`10.1088/1741-2560/14/1/016003`.

Tang, Z., Li, C., & Sun, S. (2017). Single-trial eeg classification of motor imagery using deep convolutional neural networks. *Optik*, *130*, 11–18. doi:`10.1016/j.ijleo.2016.10.117`.

470 Tolic, M., & Jovic, F. (2013). Classification of wavelet transformed EEG signals with neural network for imagined mental and motor tasks. *Kinesiology*, *45*, 130–138.

Vallabhaneni, A., & He, B. (2004). Motor imagery task classification for brain computer interface applications using spatiotemporal principle component analysis. *Neurological Research*, *26*, 282–287.

Wang, Y., Gao, S., & Gao, X. (2005). Common spatial pattern method for channel selection in motor imagery
475 based brain-computer interface. *2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, (pp. 5392–5395). doi:`10.1109/IEMBS.2005.1615701`.

World Stroke Organization (WSO) (2016). WSO background and mission statement. *Annual Report*, (p. 6).

Yang, H., Sakhavi, S., Ang, K. K., & Guan, C. (2015). On the use of convolutional neural networks and augmented CSP features for multi-class motor imagery of eeg signals classification. *IEEE Engineering in Medicine and*
480 *Biology Society Conference Proceedings*, *2015*, 2620–2623. doi:`10.1109/EMBC.2015.7318929`.

Yang, Y., Bloch, I., Chevallier, S., & Wiart, J. (2016). Subject-specific channel selection using time information for motor imagery brain–computer interfaces. *Cognitive Computation*, *8*, 505–518. doi:`10.1007/s12559-015-9379-z`.

Yang, Y., Chevallier, S., Wiart, J., & Bloch, I. (2014). Time-frequency optimization for discrimination between
485 imagination of right and left hand movements based on two bipolar electroencephalography channels. *EURASIP Journal on Advances in Signal Processing*, *38*, 1 – 18. doi:`10.1186/1687-6180-2014-38`.

Yang, Y., Chevallier, S., Wiart, J., & Bloch, I. (2017). Subject-specific time-frequency selection for multi-class motor imagery-based BCIs using few laplacian EEG channels. *Biomedical Signal Processing and Control*, *38*, 302–311.

490 Young, B. M., Williams, J., & Prabhakaran, V. (2014). BCI-FES: could a new rehabilitation device hold fresh promise for stroke patients? *Expert Review of Medical Devices*, *11*, 537–539. doi:`10.1586/17434440.2014.941811`.